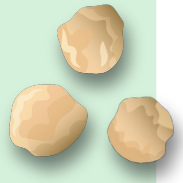
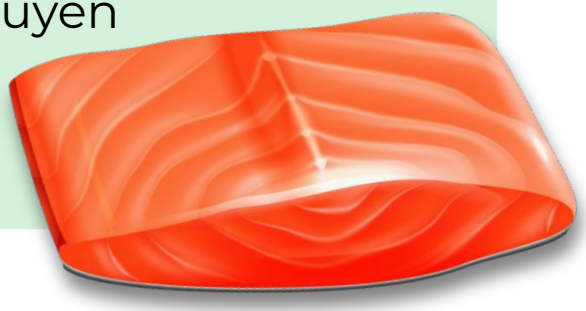
Fresh Fridge

The Food Spoilage Prevention Compartment



By Alex Paz, Michael Aquilina, Khanh Nguyen
Date: June-3-2024



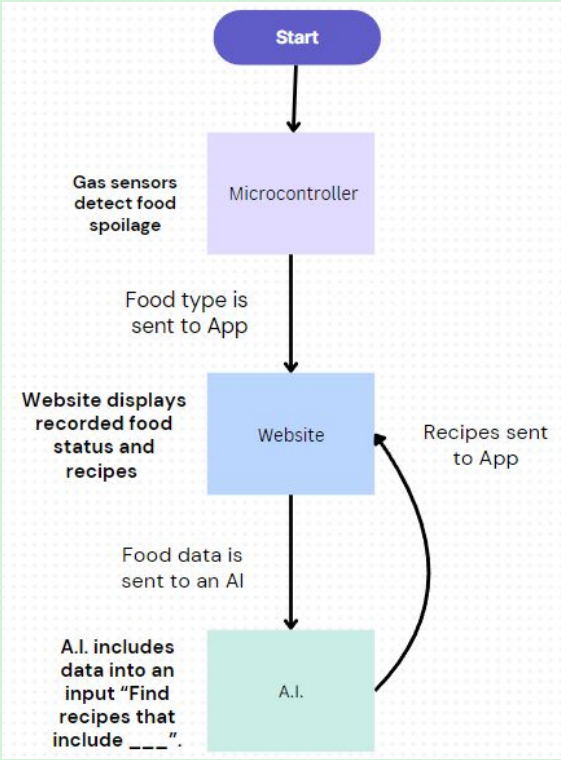
Problem Definition

Many college students and busy adults unintentionally **waste food** by forgetting about the groceries and leftovers in their refrigerators which leads to **food spoilage**.

Proposed Idea

Smart container that can detect spoiled food and identify spoiled food type using built-in **gas sensors**.
Plus an accompanying **website** that will:

- **Notify users** of detected spoilage.
- **Suggest recipes** to utilize the food before it spoils, leveraging AI-driven tips to encourage food consumption.



The Mechanics

Different foods emits different **gases** when decomposing.

- This can be used to determine what type of food is going bad

Food	**Gas Released**	**Sensors**
Milk, raw meat, fish	Ammonia	MQ-137
Banana, Fermented Food	Ethanol	MQ-5
Vegetables	Methane	MQ-4
All Foods	CO2	CO2

Testable Hypothesis

Hypothesis:

Increasing awareness of food health and providing consumption **reminders** as well as **recipe suggestions** will lead to a reduction in food waste.

Quantification:

- Determine a **difference** between healthy and unhealthy foods, utilizing sensor data
- Assess the **quality** of the suggested recipes (possibly cook them ourselves).

Milestone Timeline

Breadboard sensors with microcontroller and setup readings

Plan Arduino communication with website or app

(Week 6-7)

3D print final container and assemble hardware with tupperware

Finish website with live data, analytics, and AI recipe generator

(Week 8-9)

Predict type of food in container depending on sensor readings

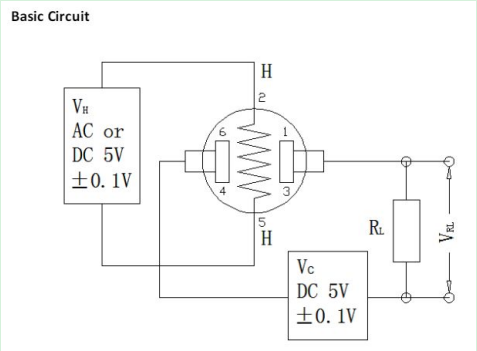
Flush out website, or migrate to a mobile app

Stretch Goals

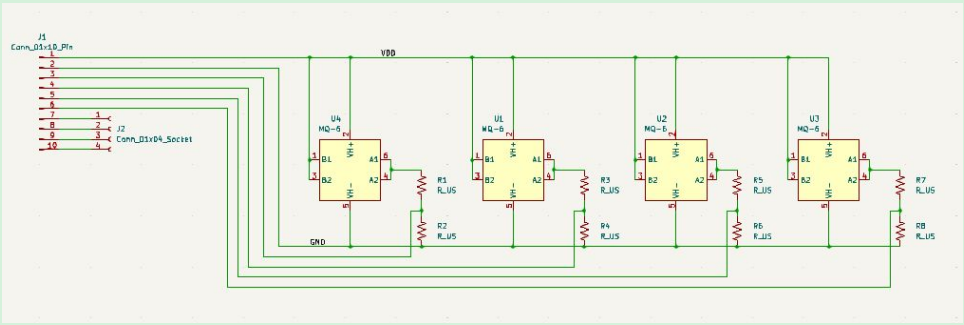
(Week 10+)

Milestones Achieved - Sensors

- Gas sensors operate on **5V input/output** and needs a 4.7k resistive load.
- Used built-in **breakout board** to step up **3.3V** to **5V** to power the entire PCB.
- **PCB** used resistor dividers to convert non-I2C voltage output



MQ-4 (Methane)



MQ-137 (Ammonia)

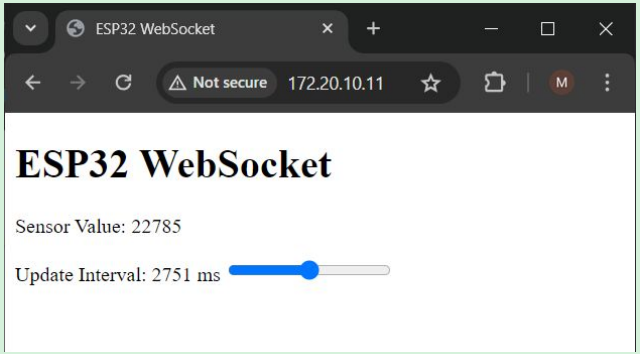
Milestones Achieved - Website

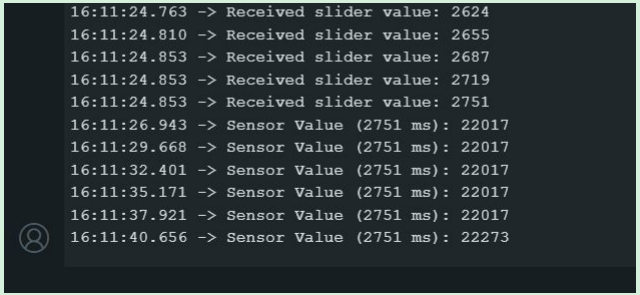
Software

Successfully got the **ESP32** to connect **wifi** and host a **webpage**.

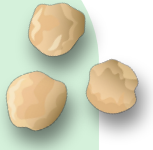
1. The **ESP32** and **webpage** have two-way communication with each other.

2. Using **websocket**, the communication is in **real-time**, and easy to set up.

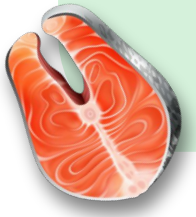


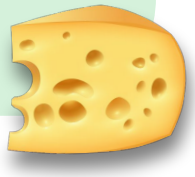


Milestone Struggles - Hardware

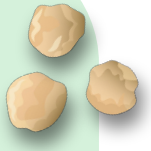


- **Part Orders**
  - 2/4 sensors (CO_2 and Ethanol) took too long to arrive
  - Didn't have enough time to gather much food data
  - Restricted food types *(bread, fruits, and vegetables)*
  -
- **Gas Sensors reliability**
  - Gas sensors are very sensitive *(temperature, operation duration, different environments)*
  - Gas sensors require a 12-hour "break-in" period to get stable readings.
  - Ethanol (MQ-5) sensor caused a brownout (omitted)





Milestone Struggles - Software

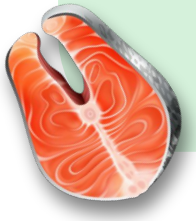


Hardware Access

- **Software** group rarely had access to hardware to test properly.
  - Needs **ESP32** to start up **websocket server**.
- **Limited** hardware access meant few opportunities to test properly
  - A testable environment without the **ESP32** was used.

Server Development

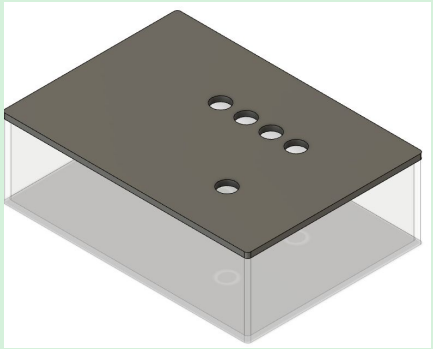
- **Lack** of **websocket** experience:
  - Did **not** know how to store HTML files **locally.**
    - Subpages would not work.
- Took a while to discover **SPIFFS File System**
  - Allowed the ESP32 to store files locally.

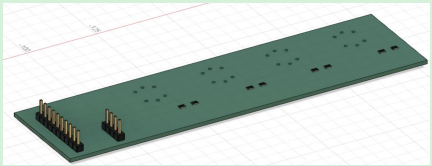


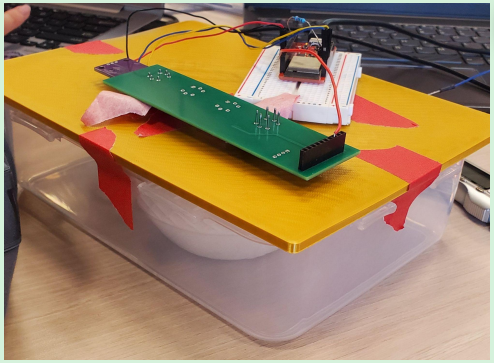
Prototype - Hardware

CAD Compartment

1. **Modeled** and **prototyped** lid by:
   a. Taking **measurements** of sensors and tupperware containers.
   b. Integrating **PCB** from KiCAD.







Assembled tupperware with 3D printed lid, ESP32, and sensors integrated with PCB

- Used for measurement/testing purposes
- Fairly airtight
- Contains food samples for testing

Prototype - Software

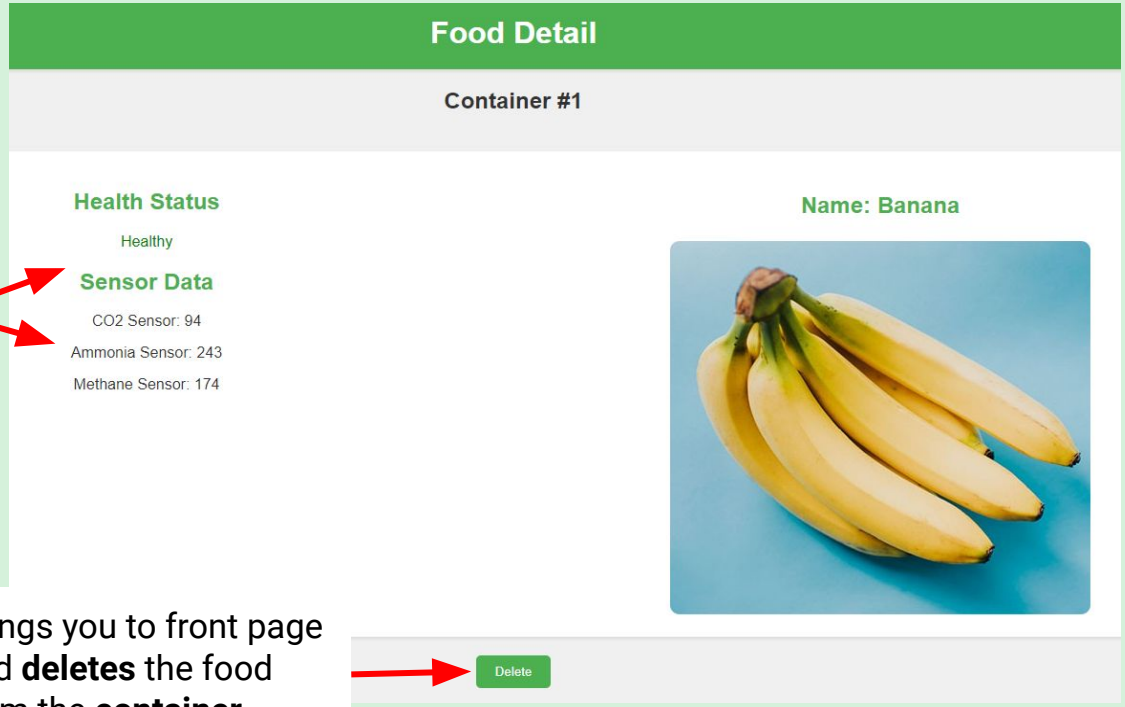
Containers take you to a generated container **page** when **clicked**.

Allows **input** for food type and selected container.

Prototype - Software

Health status **updates** when a **sensor** passes a threshold

Brings you to front page and **deletes** the food from the **container**



Prototype - Software

OpenAI

Your Food

Container 1: Banana
Container 2: Strawberry
Container 3: Apple

Generate Recipe

Recipe:

Fruit Salad Recipe:

Ingredients:
- 1 banana
- 5 strawberries
- 1 apple

Instructions:
1. Peel and slice the banana into rounds.
2. Wash and slice the strawberries into halves.
3. Wash and dice the apple into small pieces.
4. Combine all the fruits in a bowl and gently mix together.
5. Serve the fruit salad immediately or refrigerate for a refreshing and healthy snack.

- **The Goal**: Eat these foods before they expire.
- **Web page** generates a **recipe** with the ingredients in your containers using **OpenAI API**.

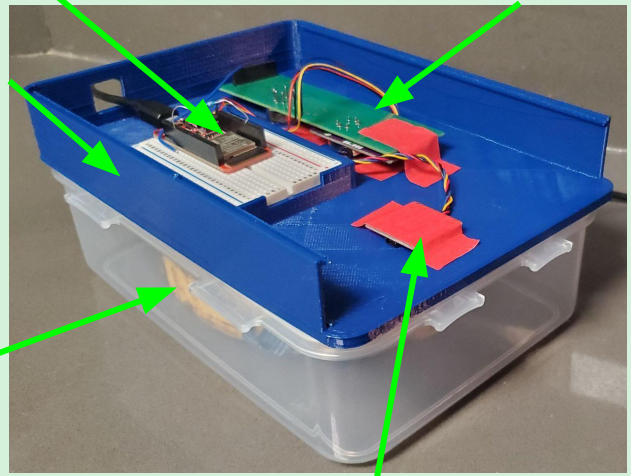
The Result: FreshFridge makes it **easy** to find these recipes and **motivates** you to eat the food in your containers before they spoil!

Final Design

Microcontroller (Sparkfun ESP32)

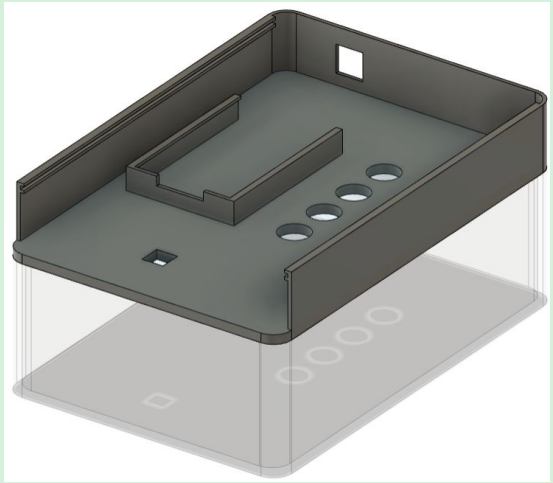
PCB with Methane and Ammonia Sensors

3D-printed Lid



Food

CO2/Humidity/Temperature Sensor



Final Lid Design in CAD

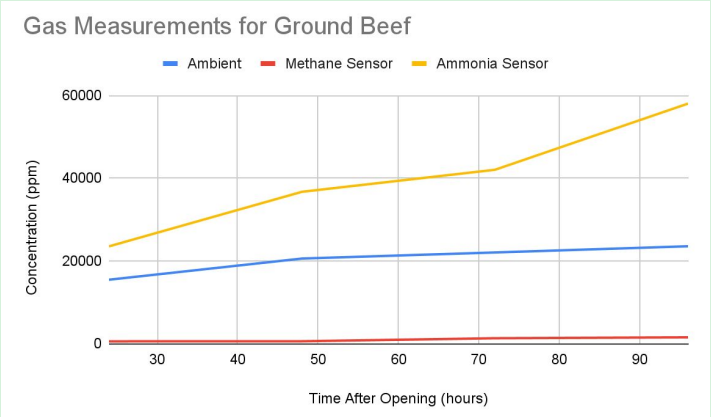
Trial and Results

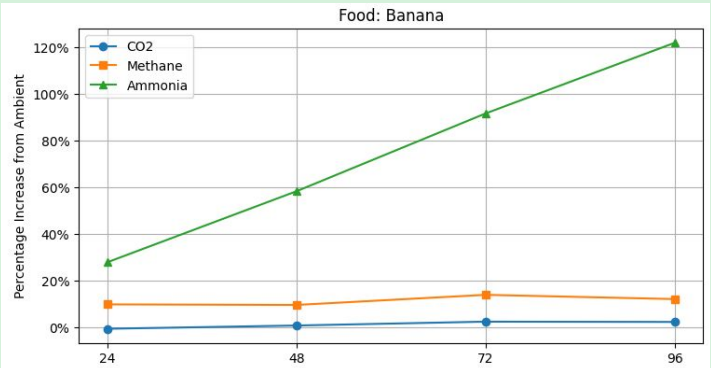
Tested to see if certain sensors can pick up certain gases better than others.

- **Method:** measured Methane and Ammonia gas concentrations of **ground beef** and **bananas** over a period of 4 days (96 hours).

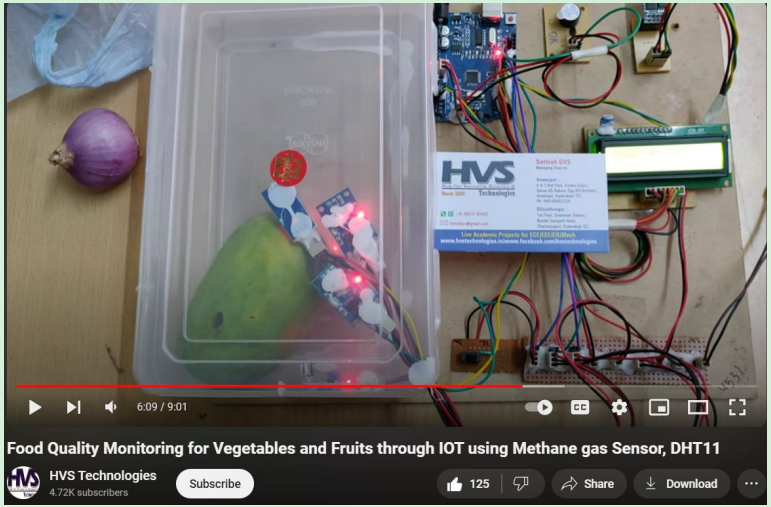
Results:

- Ammonia levels **increased proportionally** to the hours left out
- Methane levels barely changed.
- Ambient levels went up the more measurements we did (residues in sensors?)

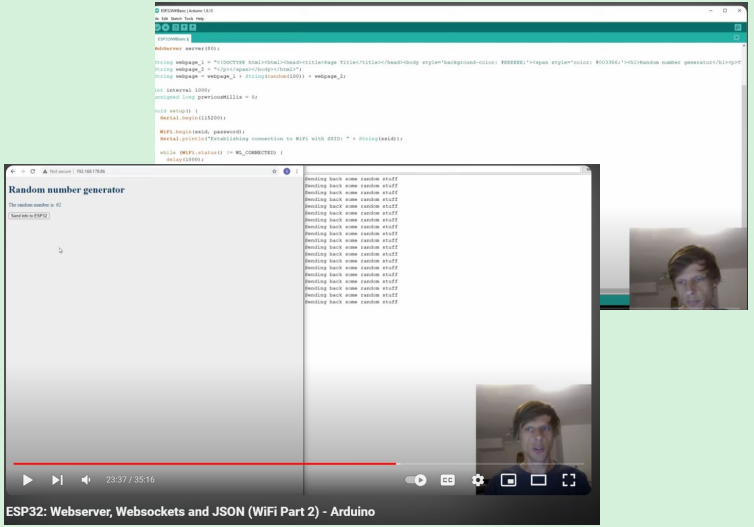




Notable Resources



The video that first showed us the potential of gas sensors to detect food spoilage. [Source]



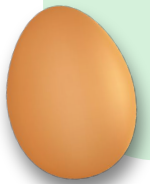
Helpful tutorial on Websocket Servers for our webpage. [Source]

References, Citations, & Helpful Resources

Study	Study explaining biogas emission of composting foods
Video	Usage of gas sensors and microcontrollers to detect food spoilage.
Video	Demonstrates how to get gas concentration from sensor readings.
Video	OpenAI tutorial for making API requests in Javascript
Video	ESP32: Webserver, Websockets and JSON- Arduino








Thank you